

Aufgabe 8.1 (4 Punkte)

Gib eine Folge von n Operationen auf einer sortierten Liste an, die bei der Implementierung mit (2,3)-Bäumen $\Omega(n \log n)$ Knotenspaltungen und -verschmelzungen auslöst.

Aufgabe 8.2 (4 Punkte)

In der Vorlesung wurde gezeigt, dass beim Inkrementieren eines Binärzählers amortisiert konstant viele Bits umgesetzt werden. Jetzt möchten wir den Binärzähler nicht nur inkrementieren, sondern auch zurück auf 0 setzen können (resetten). Beschreibe, wie man diesen Binärzähler implementieren kann, sodass jede Folge von n inkrementier- und reset-Operationen $O(n)$ Zeit benötigt, wenn der Zähler am Anfang den Wert 0 enthält. Beweise diese Schranke mit der Bankkontomethode.

Hinweis: Speichere einen Zeiger auf das höchste Bit, das eine Eins enthält.

Aufgabe 8.3 (4 Punkte)

Betrachte einen binären Suchbaum mit Höhe h der erweitert ist mit dem Feld $t.size$, wie in der Vorlesung gezeigt.

Erkläre, wie man zu gegebenem Schlüssel k in $O(h)$ Zeit den Rang des Folgeneintrags mit Schlüssel k bestimmt.